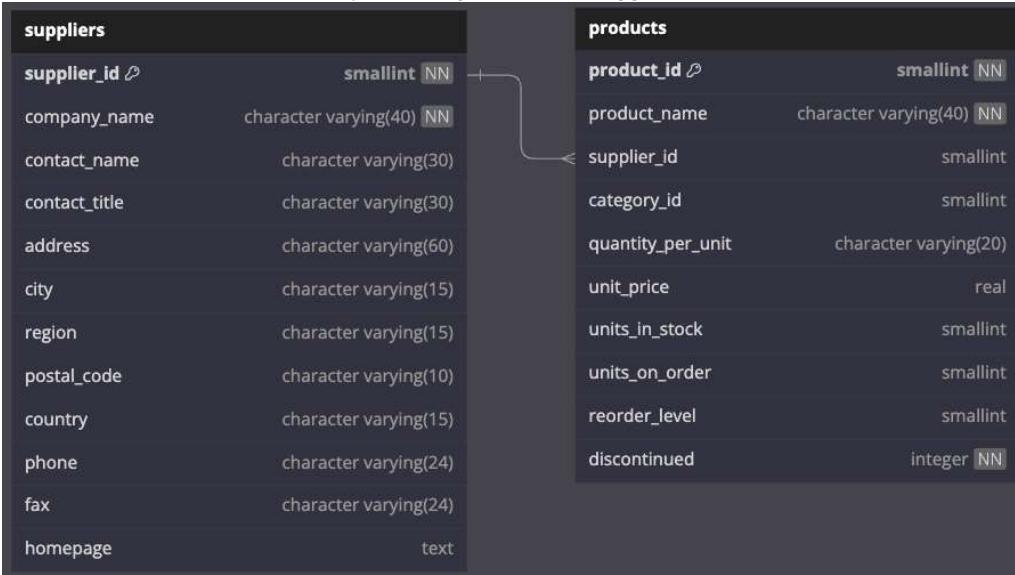


Питання 1

Правильно

Балів 1,00 з 1,00

Here's a data model of the relationship between **products** and **suppliers**.



We'd like to show, for each product, the associated supplier from *Germany* and *Spain*. Show the **product_id**, **product_name**, and the **company_name** of the **suppliers**. Sort by **product_id**.

Note. The Join clause is used to join two or more relational database tables together in a logical way.

For example:

Test	Result		
-- Testing with original db	product_id	product_name	company_name
	-----	-----	-----
	11	Queso Cabrales	Cooperativa de Quesos 'Las Cabras'
	12	Queso Manchego La Pastora	Cooperativa de Quesos 'Las Cabras'
	25	NuNuCa Nuu-Nougat-Creme	Heli Suuwaren GmbH & Co. KG
	26	Gumbur Gummiburchen	Heli Suuwaren GmbH & Co. KG
	27	Schoggi Schokolade	Heli Suuwaren GmbH & Co. KG
	28	Russle Sauerkraut	Plutzer Lebensmittelgroumukte AG
	29	Thuringer Rostbratwurst	Plutzer Lebensmittelgroumukte AG
	30	Nord-Ost Matjeshering	Nord-Ost-Fisch Handelsgesellschaft mbH
	64	Wimmers gute Semmelknudel	Plutzer Lebensmittelgroumukte AG
	75	Rhunbruu Klosterbier	Plutzer Lebensmittelgroumukte AG
	77	Original Frankfurter grune Soue	Plutzer Lebensmittelgroumukte AG

Відповідь: (penalty regime: 0 %)

```

1 | SELECT
2 |   p.product_id,
3 |   p.product_name,
4 |   s.company_name
5 | FROM products AS p
6 | JOIN suppliers AS s ON p.supplier_id = s.supplier_id
7 | WHERE s.country IN ('Germany', 'Spain')
8 | ORDER BY p.product_id;
9 |

```

	Test	Expected	Got	
<input checked="" type="checkbox"/>	-- Testing with original db	product_id product_name company_name ----- ----- 11 Queso Cabrales Cooperativa de Quesos 'Las Cabras' 12 Queso Manchego La Pastora Cooperativa de Quesos 'Las Cabras' 25 NuNuCa Nuu-Nougat-Creme Heli Suuwaren GmbH & Co. KG 26 Gumbur Gummiburchen Heli Suuwaren GmbH & Co. KG 27 Schoggi Schokolade Heli Suuwaren GmbH & Co. KG 28 Russle Sauerkraut Plutzer Lebensmittelgroumukte AG 29 Thuringer Rostbratwurst Plutzer Lebensmittelgroumukte AG 30 Nord-Ost Matjeshering Nord-Ost-Fisch Handelsgesellschaft mbH 64 Wimmers gute Semmelknudel Plutzer Lebensmittelgroumukte AG 75 Rhunbruu Klosterbier Plutzer Lebensmittelgroumukte AG 77 Original Frankfurter grune Soue Plutzer Lebensmittelgroumukte AG	product_id product_name company_name ----- ----- 11 Queso Cabrales Cooperativa de Quesos 'Las Cabras' 12 Queso Manchego La Pastora Cooperativa de Quesos 'Las Cabras' 25 NuNuCa Nuu-Nougat-Creme Heli Suuwaren GmbH & Co. KG 26 Gumbur Gummiburchen Heli Suuwaren GmbH & Co. KG 27 Schoggi Schokolade Heli Suuwaren GmbH & Co. KG 28 Russle Sauerkraut Plutzer Lebensmittelgroumukte AG 29 Thuringer Rostbratwurst Plutzer Lebensmittelgroumukte AG 30 Nord-Ost Matjeshering Nord-Ost-Fisch Handelsgesellschaft mbH 64 Wimmers gute Semmelknudel Plutzer Lebensmittelgroumukte AG 75 Rhunbruu Klosterbier Plutzer Lebensmittelgroumukte AG 77 Original Frankfurter grune Soue Plutzer Lebensmittelgroumukte AG	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	-- Testing with extra row	product_id product_name company_name ----- ----- 25 NuNuCa Nuu-Nougat-Creme Heli Suuwaren GmbH & Co. KG 26 Gumbur Gummiburchen Heli Suuwaren GmbH & Co. KG 27 Schoggi Schokolade Heli Suuwaren GmbH & Co. KG 28 Russle Sauerkraut Plutzer Lebensmittelgroumukte AG 29 Thuringer Rostbratwurst Plutzer Lebensmittelgroumukte AG 30 Nord-Ost Matjeshering Nord-Ost-Fisch Handelsgesellschaft mbH 64 Wimmers gute Semmelknudel Plutzer Lebensmittelgroumukte AG 75 Rhunbruu Klosterbier Plutzer Lebensmittelgroumukte AG 77 Original Frankfurter grune Soue Plutzer Lebensmittelgroumukte AG	product_id product_name company_name ----- ----- 25 NuNuCa Nuu-Nougat-Creme Heli Suuwaren GmbH & Co. KG 26 Gumbur Gummiburchen Heli Suuwaren GmbH & Co. KG 27 Schoggi Schokolade Heli Suuwaren GmbH & Co. KG 28 Russle Sauerkraut Plutzer Lebensmittelgroumukte AG 29 Thuringer Rostbratwurst Plutzer Lebensmittelgroumukte AG 30 Nord-Ost Matjeshering Nord-Ost-Fisch Handelsgesellschaft mbH 64 Wimmers gute Semmelknudel Plutzer Lebensmittelgroumukte AG 75 Rhunbruu Klosterbier Plutzer Lebensmittelgroumukte AG 77 Original Frankfurter grune Soue Plutzer Lebensmittelgroumukte AG	<input checked="" type="checkbox"/>

Пройшов усі тести! Правильно

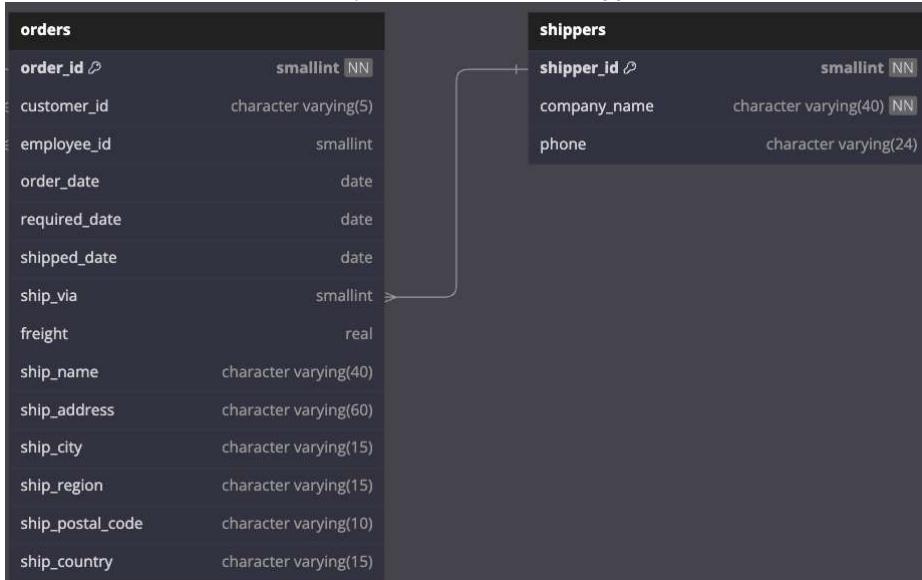
Бали за цю відповідь: 1,00/1,00.

Питання 2

Правильно

Балів 1,00 з 1,00

Here's a data model of the relationship between **orders** and **shippers**.



We'd like to show a list of the orders that were made, including the shipper that was used. Show the **order_id**, **order_date**, and **company_name** of the **shippers**, and sort by **order_id**. Show only those rows with an **order_id** of less than 10260.

For example:

Test	Result
-- Testing with original db	<pre> order_id order_date company_name ----- ----- 10248 2014-06-29 Federal Shipping 10249 2014-07-24 Speedy Express 10250 2014-07-13 United Package 10251 2014-07-13 Speedy Express 10252 2014-07-14 United Package 10253 2014-07-01 United Package 10254 2014-07-16 United Package 10255 2014-07-17 Federal Shipping 10256 2014-07-20 United Package 10257 2014-07-21 Federal Shipping 10258 2014-07-22 Speedy Express 10259 2014-07-23 Federal Shipping </pre>

Відповідь: (penalty regime: 0 %)

```

1 | SELECT
2 |     o.order_id,
3 |     o.order_date,
4 |     s.company_name
5 | FROM orders AS o
6 | JOIN shippers AS s ON o.ship_via = s.shipper_id
7 | WHERE o.order_id < 10260
8 | ORDER BY o.order_id;
9 |

```

	Test	Expected	Got																																																																															
✓	-- Testing with original db	<table border="1"> <thead> <tr> <th>order_id</th> <th>order_date</th> <th>company_name</th> </tr> </thead> <tbody> <tr><td>10248</td><td>2014-06-29</td><td>Federal Shipping</td></tr> <tr><td>10249</td><td>2014-07-24</td><td>Speedy Express</td></tr> <tr><td>10250</td><td>2014-07-13</td><td>United Package</td></tr> <tr><td>10251</td><td>2014-07-13</td><td>Speedy Express</td></tr> <tr><td>10252</td><td>2014-07-14</td><td>United Package</td></tr> <tr><td>10253</td><td>2014-07-01</td><td>United Package</td></tr> <tr><td>10254</td><td>2014-07-16</td><td>United Package</td></tr> <tr><td>10255</td><td>2014-07-17</td><td>Federal Shipping</td></tr> <tr><td>10256</td><td>2014-07-20</td><td>United Package</td></tr> <tr><td>10257</td><td>2014-07-21</td><td>Federal Shipping</td></tr> <tr><td>10258</td><td>2014-07-22</td><td>Speedy Express</td></tr> <tr><td>10259</td><td>2014-07-23</td><td>Federal Shipping</td></tr> </tbody> </table>	order_id	order_date	company_name	10248	2014-06-29	Federal Shipping	10249	2014-07-24	Speedy Express	10250	2014-07-13	United Package	10251	2014-07-13	Speedy Express	10252	2014-07-14	United Package	10253	2014-07-01	United Package	10254	2014-07-16	United Package	10255	2014-07-17	Federal Shipping	10256	2014-07-20	United Package	10257	2014-07-21	Federal Shipping	10258	2014-07-22	Speedy Express	10259	2014-07-23	Federal Shipping	<table border="1"> <thead> <tr> <th>order_id</th> <th>order_date</th> <th>company_name</th> </tr> </thead> <tbody> <tr><td>10248</td><td>2014-06-29</td><td>Federal Shipping</td></tr> <tr><td>10249</td><td>2014-07-24</td><td>Speedy Express</td></tr> <tr><td>10250</td><td>2014-07-13</td><td>United Package</td></tr> <tr><td>10251</td><td>2014-07-13</td><td>Speedy Express</td></tr> <tr><td>10252</td><td>2014-07-14</td><td>United Package</td></tr> <tr><td>10253</td><td>2014-07-01</td><td>United Package</td></tr> <tr><td>10254</td><td>2014-07-16</td><td>United Package</td></tr> <tr><td>10255</td><td>2014-07-17</td><td>Federal Shipping</td></tr> <tr><td>10256</td><td>2014-07-20</td><td>United Package</td></tr> <tr><td>10257</td><td>2014-07-21</td><td>Federal Shipping</td></tr> <tr><td>10258</td><td>2014-07-22</td><td>Speedy Express</td></tr> <tr><td>10259</td><td>2014-07-23</td><td>Federal Shipping</td></tr> </tbody> </table>	order_id	order_date	company_name	10248	2014-06-29	Federal Shipping	10249	2014-07-24	Speedy Express	10250	2014-07-13	United Package	10251	2014-07-13	Speedy Express	10252	2014-07-14	United Package	10253	2014-07-01	United Package	10254	2014-07-16	United Package	10255	2014-07-17	Federal Shipping	10256	2014-07-20	United Package	10257	2014-07-21	Federal Shipping	10258	2014-07-22	Speedy Express	10259	2014-07-23	Federal Shipping	✓
order_id	order_date	company_name																																																																																
10248	2014-06-29	Federal Shipping																																																																																
10249	2014-07-24	Speedy Express																																																																																
10250	2014-07-13	United Package																																																																																
10251	2014-07-13	Speedy Express																																																																																
10252	2014-07-14	United Package																																																																																
10253	2014-07-01	United Package																																																																																
10254	2014-07-16	United Package																																																																																
10255	2014-07-17	Federal Shipping																																																																																
10256	2014-07-20	United Package																																																																																
10257	2014-07-21	Federal Shipping																																																																																
10258	2014-07-22	Speedy Express																																																																																
10259	2014-07-23	Federal Shipping																																																																																
order_id	order_date	company_name																																																																																
10248	2014-06-29	Federal Shipping																																																																																
10249	2014-07-24	Speedy Express																																																																																
10250	2014-07-13	United Package																																																																																
10251	2014-07-13	Speedy Express																																																																																
10252	2014-07-14	United Package																																																																																
10253	2014-07-01	United Package																																																																																
10254	2014-07-16	United Package																																																																																
10255	2014-07-17	Federal Shipping																																																																																
10256	2014-07-20	United Package																																																																																
10257	2014-07-21	Federal Shipping																																																																																
10258	2014-07-22	Speedy Express																																																																																
10259	2014-07-23	Federal Shipping																																																																																
✓	-- Testing with extra row	<table border="1"> <thead> <tr> <th>order_id</th> <th>order_date</th> <th>company_name</th> </tr> </thead> <tbody> <tr><td>10248</td><td>2014-06-29</td><td>Speedy Express</td></tr> <tr><td>10249</td><td>2014-07-24</td><td>Speedy Express</td></tr> <tr><td>10250</td><td>2014-07-13</td><td>United Package</td></tr> <tr><td>10251</td><td>2014-07-13</td><td>Speedy Express</td></tr> <tr><td>10252</td><td>2014-07-14</td><td>United Package</td></tr> <tr><td>10253</td><td>2014-07-01</td><td>United Package</td></tr> <tr><td>10254</td><td>2014-07-16</td><td>United Package</td></tr> <tr><td>10255</td><td>2014-07-17</td><td>Federal Shipping</td></tr> <tr><td>10256</td><td>2014-07-20</td><td>United Package</td></tr> <tr><td>10257</td><td>2014-07-21</td><td>Federal Shipping</td></tr> <tr><td>10258</td><td>2014-07-22</td><td>Speedy Express</td></tr> <tr><td>10259</td><td>2014-07-23</td><td>Federal Shipping</td></tr> </tbody> </table>	order_id	order_date	company_name	10248	2014-06-29	Speedy Express	10249	2014-07-24	Speedy Express	10250	2014-07-13	United Package	10251	2014-07-13	Speedy Express	10252	2014-07-14	United Package	10253	2014-07-01	United Package	10254	2014-07-16	United Package	10255	2014-07-17	Federal Shipping	10256	2014-07-20	United Package	10257	2014-07-21	Federal Shipping	10258	2014-07-22	Speedy Express	10259	2014-07-23	Federal Shipping	<table border="1"> <thead> <tr> <th>order_id</th> <th>order_date</th> <th>company_name</th> </tr> </thead> <tbody> <tr><td>10248</td><td>2014-06-29</td><td>Speedy Express</td></tr> <tr><td>10249</td><td>2014-07-24</td><td>Speedy Express</td></tr> <tr><td>10250</td><td>2014-07-13</td><td>United Package</td></tr> <tr><td>10251</td><td>2014-07-13</td><td>Speedy Express</td></tr> <tr><td>10252</td><td>2014-07-14</td><td>United Package</td></tr> <tr><td>10253</td><td>2014-07-01</td><td>United Package</td></tr> <tr><td>10254</td><td>2014-07-16</td><td>United Package</td></tr> <tr><td>10255</td><td>2014-07-17</td><td>Federal Shipping</td></tr> <tr><td>10256</td><td>2014-07-20</td><td>United Package</td></tr> <tr><td>10257</td><td>2014-07-21</td><td>Federal Shipping</td></tr> <tr><td>10258</td><td>2014-07-22</td><td>Speedy Express</td></tr> <tr><td>10259</td><td>2014-07-23</td><td>Federal Shipping</td></tr> </tbody> </table>	order_id	order_date	company_name	10248	2014-06-29	Speedy Express	10249	2014-07-24	Speedy Express	10250	2014-07-13	United Package	10251	2014-07-13	Speedy Express	10252	2014-07-14	United Package	10253	2014-07-01	United Package	10254	2014-07-16	United Package	10255	2014-07-17	Federal Shipping	10256	2014-07-20	United Package	10257	2014-07-21	Federal Shipping	10258	2014-07-22	Speedy Express	10259	2014-07-23	Federal Shipping	✓
order_id	order_date	company_name																																																																																
10248	2014-06-29	Speedy Express																																																																																
10249	2014-07-24	Speedy Express																																																																																
10250	2014-07-13	United Package																																																																																
10251	2014-07-13	Speedy Express																																																																																
10252	2014-07-14	United Package																																																																																
10253	2014-07-01	United Package																																																																																
10254	2014-07-16	United Package																																																																																
10255	2014-07-17	Federal Shipping																																																																																
10256	2014-07-20	United Package																																																																																
10257	2014-07-21	Federal Shipping																																																																																
10258	2014-07-22	Speedy Express																																																																																
10259	2014-07-23	Federal Shipping																																																																																
order_id	order_date	company_name																																																																																
10248	2014-06-29	Speedy Express																																																																																
10249	2014-07-24	Speedy Express																																																																																
10250	2014-07-13	United Package																																																																																
10251	2014-07-13	Speedy Express																																																																																
10252	2014-07-14	United Package																																																																																
10253	2014-07-01	United Package																																																																																
10254	2014-07-16	United Package																																																																																
10255	2014-07-17	Federal Shipping																																																																																
10256	2014-07-20	United Package																																																																																
10257	2014-07-21	Federal Shipping																																																																																
10258	2014-07-22	Speedy Express																																																																																
10259	2014-07-23	Federal Shipping																																																																																

Пройшов усі тести!

Правильно

Бали за цю відповідь: 1,00/1,00.

Питання 3

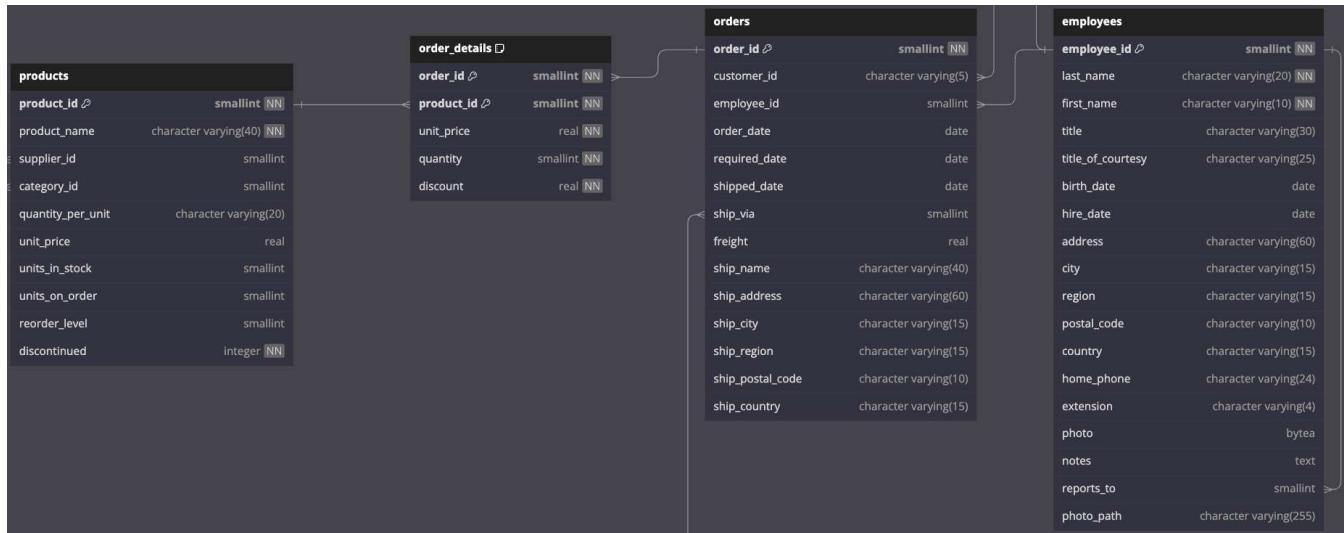
Правильно

Балів 1,00 з 1,00

We're doing inventory, and need to show information about **order_id**, a list of products, and their quantity for orders which were shipped by employee *Leverling Janet* with quantities greater than 50.

The result should be sorted by **quantity**.

Use the model:



For example:

Test	Result		
-- Testing with original db	order_id	product_name	quantity
	-----	-----	-----
	10273	Boston Crab Meat	60
	10442	Louisiana Hot Spiced Okra	60
	10479	Raclette Courdavault	60
	10492	NuNuCa Nuu-Nougat-Creme	60
	10540	Aniseed Syrup	60
	10547	Inlagd Sill	60
	10570	Gnocchi di nonna Alice	60
	10638	Mozzarella di Giovanni	60
	10693	Tourtire	60
	10700	Flotemysost	60
	10758	Filo Mix	60
	10762	Gnocchi di nonna Alice	60
	10817	Boston Crab Meat	60
	10918	Chai	60
	10988	Uncle Bob's Organic Dried Pears	60
	11021	Gumbur Gummiburchen	63
	10854	Konbu	65
	10514	Gnocchi di nonna Alice	70
	10442	Tourtire	80
	10765	Louisiana Fiery Hot Pepper Sauce	80
	10897	Thuringer Rostbratwurst	80
	10895	Boston Crab Meat	91
	10854	Ikura	100
	10895	Camembert Pierrot	100
	10895	Guaranu Fantustica	110

Відповідь: (penalty regime: 0 %)

```

1 | SELECT
2 |   o.order_id,
3 |   p.product_name,
4 |   od.quantity
5 | FROM orders AS o
6 | JOIN order_details AS od ON o.order_id = od.order_id
7 | JOIN products AS p ON od.product_id = p.product_id
  
```

```

8 | JOIN employees AS e ON o.employee_id = e.employee_id
9 | WHERE e.first_name = 'Janet'
10 |   AND e.last_name = 'Leverling'
11 |   AND od.quantity > 50
12 | ORDER BY od.quantity;
13 |

```

[]

	Tect	Expected	Got	
<input checked="" type="checkbox"/>	-- Testing with original db	order_id product_name quantity ----- ----- 10273 Boston Crab Meat 60 10442 Louisiana Hot Spiced Okra 60 10479 Raclette Courdavault 60 10492 NuNuCa Nuu-Nougat-Creme 60 10540 Aniseed Syrup 60 10547 Inlagd Sill 60 10570 Gnocchi di nonna Alice 60 10638 Mozzarella di Giovanni 60 10693 Tourtiure 60 10700 Flotemysost 60 10758 Filo Mix 60 10762 Gnocchi di nonna Alice 60 10817 Boston Crab Meat 60 10918 Chai 60 10988 Uncle Bob's Organic Dried Pears 60 11021 Gumbur Gummiburchen 63 10854 Konbu 65 10514 Gnocchi di nonna Alice 70 10442 Tourtiure 80 10765 Louisiana Fiery Hot Pepper Sauce 80 10897 Thuringer Rostbratwurst 80 10895 Boston Crab Meat 91 10854 Ikura 100 10895 Camembert Pierrot 100 10895 Guaranu Fantustica 110	order_id product_name quantity ----- ----- 10273 Boston Crab Meat 60 10442 Louisiana Hot Spiced Okra 60 10479 Raclette Courdavault 60 10492 NuNuCa Nuu-Nougat-Creme 60 10540 Aniseed Syrup 60 10547 Inlagd Sill 60 10570 Gnocchi di nonna Alice 60 10638 Mozzarella di Giovanni 60 10693 Tourtiure 60 10700 Flotemysost 60 10758 Filo Mix 60 10762 Gnocchi di nonna Alice 60 10817 Boston Crab Meat 60 10918 Chai 60 10988 Uncle Bob's Organic Dried Pears 60 11021 Gumbur Gummiburchen 63 10854 Konbu 65 10514 Gnocchi di nonna Alice 70 10442 Tourtiure 80 10765 Louisiana Fiery Hot Pepper Sauce 80 10897 Thuringer Rostbratwurst 80 10895 Boston Crab Meat 91 10854 Ikura 100 10895 Camembert Pierrot 100 10895 Guaranu Fantustica 110	<input checked="" type="checkbox"/>

	Test	Expected	Got	
✓	-- Testing with extra row	order_id product_name quantity ----- ----- 10442 Louisiana Hot Spiced Okra 60 10479 Raclette Courdavault 60 10492 NuNuCa Nuu-Nougat-Creme 60 10540 Aniseed Syrup 60 10547 Inlagd Sill 60 10570 Gnocchi di nonna Alice 60 10638 Mozzarella di Giovanni 60 10693 Tourtiure 60 10700 Flotemysost 60 10758 Filo Mix 60 10762 Gnocchi di nonna Alice 60 10817 Boston Crab Meat 60 10918 Chai 60 10988 Uncle Bob's Organic Dried Pears 60 11021 Gumbur Gummiburchen 63 10854 Konbu 65 10514 Gnocchi di nonna Alice 70 10442 Tourtiure 80 10765 Louisiana Fiery Hot Pepper Sauce 80 10897 Thuringer Rostbratwurst 80 10895 Boston Crab Meat 91 10854 Ikura 100 10895 Camembert Pierrot 100 10895 Guaranu Fantustica 110	order_id product_name quantity ----- ----- 10442 Louisiana Hot Spiced Okra 60 10479 Raclette Courdavault 60 10492 NuNuCa Nuu-Nougat-Creme 60 10540 Aniseed Syrup 60 10547 Inlagd Sill 60 10570 Gnocchi di nonna Alice 60 10638 Mozzarella di Giovanni 60 10693 Tourtiure 60 10700 Flotemysost 60 10758 Filo Mix 60 10762 Gnocchi di nonna Alice 60 10817 Boston Crab Meat 60 10918 Chai 60 10988 Uncle Bob's Organic Dried Pears 60 11021 Gumbur Gummiburchen 63 10854 Konbu 65 10514 Gnocchi di nonna Alice 70 10442 Tourtiure 80 10765 Louisiana Fiery Hot Pepper Sauce 80 10897 Thuringer Rostbratwurst 80 10895 Boston Crab Meat 91 10854 Ikura 100 10895 Camembert Pierrot 100 10895 Guaranu Fantustica 110	✓

Пройшов усі тести! ✓

Правильно

Бали за цю відповідь: 1,00/1,00.

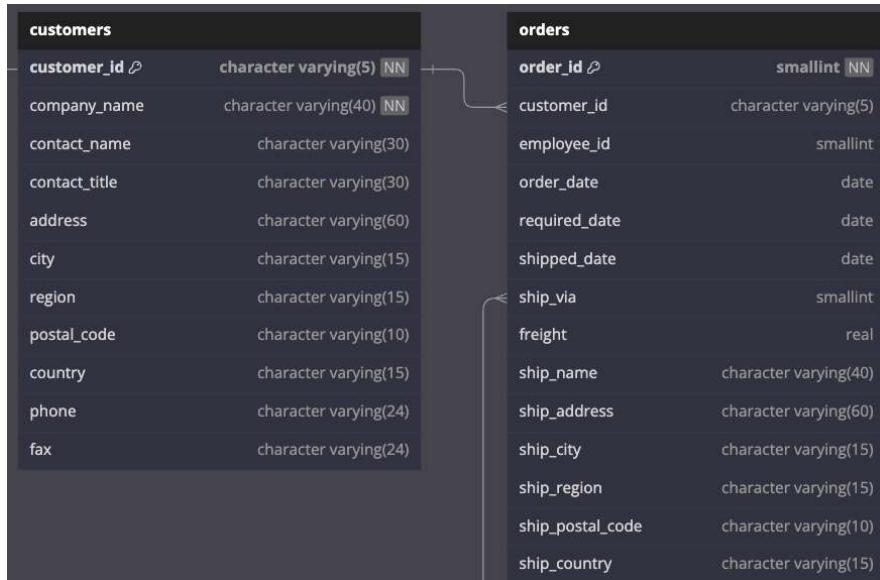
Питання 4

Правильно

Балів 1,00 з 1,00

There are some **customers** who have never actually placed an order. Show **company_name** of these customers.

Use the model:



Note. One way of doing this is to use a left join, also known as a left outer join.

For example:

Test	Result
-- Testing with original db	<pre>company_name ----- FISSA Fabrica Inter. Salchichas S.A. Paris spacialitus Roomblo Fivebridge Fadeo Zooxo Thoughtbeat Fadeo Brightbean Pixonyx Ainyx Dablist</pre>

Відповідь: (penalty regime: 0 %)

```

1 | SELECT c.company_name
2 | FROM customers AS c
3 | LEFT JOIN orders AS o ON c.customer_id = o.customer_id
4 | WHERE o.order_id IS NULL;
5 |
  
```

	Test	Expected	Got	
✓	-- Testing with original db	company_name ----- FISSA Fabrica Inter. Salchichas S.A. Paris spacialitus Roombo Fivebridge Fadeo Zooxo Thoughtbeat Fadeo Brightbean Pixonyx Ainyx Dablist	company_name ----- FISSA Fabrica Inter. Salchichas S.A. Paris spacialitus Roombo Fivebridge Fadeo Zooxo Thoughtbeat Fadeo Brightbean Pixonyx Ainyx Dablist	✓
✓	-- Testing with extra row	company_name ----- Paris spacialitus Roombo Fivebridge Fadeo Zooxo Thoughtbeat Fadeo Brightbean Pixonyx Ainyx Dablist	company_name ----- Paris spacialitus Roombo Fivebridge Fadeo Zooxo Thoughtbeat Fadeo Brightbean Pixonyx Ainyx Dablist	✓

Пройшов усі тести! ✓

Правильно

Бали за цю відповідь: 1,00/1,00.

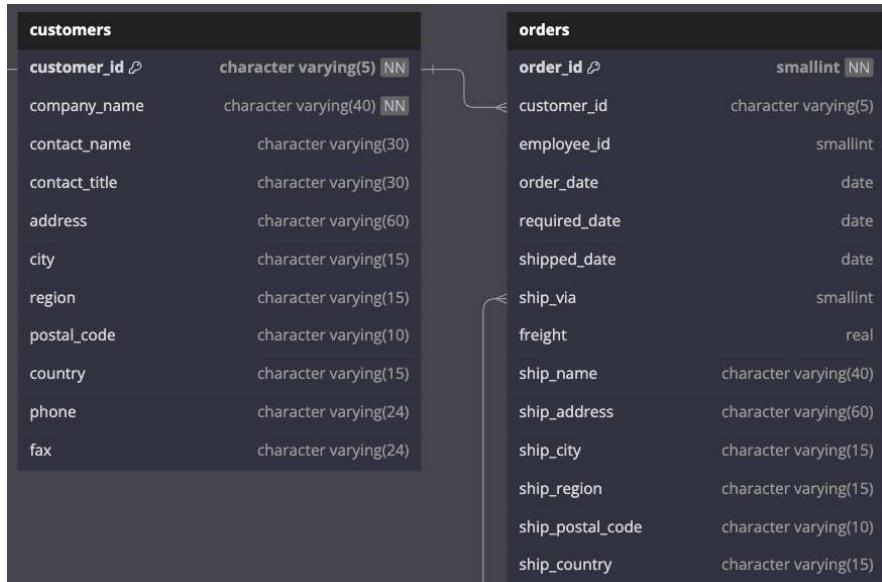
Питання 5

Правильно

Балів 1,00 з 1,00

One employee (*Margaret Peacock*, **employee_id** = 4) has placed the most orders. However, there are some customers who've never placed an order with her. Show **company_name** of only those customers who have never placed an order with her.

Use the model:



Note. One way of doing this is to use one type of outer join (left or right). Because the filters in the WHERE clause are applied after the results of the JOIN, we need the `employee_id = 4` filter in the JOIN clause, instead of the WHERE clause.

For example:

Test	Result
-- Testing with original db	<pre>company_name ----- Consolidated Holdings Du monde entier FISSA Fabrica Inter. Salchichas S.A. France restauration GROSELLA-Restaurante Laughing Bacchus Wine Cellars Lazy K Kountry Store North/South Paris spcialitus Pericles Comidas clusicas Princesa Isabel Vinhos Santu Gourmet Seven Seas Imports Spacialitus du monde The Big Cheese Vins et alcools Chevalier Roombo Fivebridge Fadeo Zooxo Thoughtbeat Fadeo Brightbean Pixonyx Ainyx Dablist</pre>

Відповідь: (penalty regime: 0 %)

```
1 | SELECT c.company_name
2 | FROM customers AS c
3 | LEFT JOIN orders AS o ON c.customer_id = o.customer_id
```

```

4 | AND o.employee_id = 4
5 | WHERE o.order_id IS NULL;
6 |

```



	Tect	Expected	Got	
<input checked="" type="checkbox"/>	-- Testing with original db	company_name ----- Consolidated Holdings Du monde entier FISSA Fabrica Inter. Salchichas S.A. France restauration GROSELLA-Restaurante Laughing Bacchus Wine Cellars Lazy K Kountry Store North/South Paris spacialitus Pericles Comidas clusicas Princesa Isabel Vinhos Santu Gourmet Seven Seas Imports Spacialitus du monde The Big Cheese Vins et alcools Chevalier Roombo Fivebridge Fadeo Zooxo Thoughtbeat Fadeo Brightbean Pixonyx Ainyx Dablist	company_name ----- Consolidated Holdings Du monde entier FISSA Fabrica Inter. Salchichas S.A. France restauration GROSELLA-Restaurante Laughing Bacchus Wine Cellars Lazy K Kountry Store North/South Paris spacialitus Pericles Comidas clusicas Princesa Isabel Vinhos Santu Gourmet Seven Seas Imports Spacialitus du monde The Big Cheese Vins et alcools Chevalier Roombo Fivebridge Fadeo Zooxo Thoughtbeat Fadeo Brightbean Pixonyx Ainyx Dablist	<input checked="" type="checkbox"/>

	Тест	Expected	Got	
✓	-- Testing with extra row	company_name ----- Du monde entier FISSA Fabrica Inter. Salchichas S.A. France restauration GROSELLA-Restaurante Laughing Bacchus Wine Cellars Lazy K Kountry Store North/South Paris spacialitus Pericles Comidas clusicas Princesa Isabel Vinhos Santu Gourmet Seven Seas Imports Spacialitus du monde The Big Cheese Vins et alcools Chevalier Roombo Fivebridge Fadeo Zooxo Thoughtbeat Fadeo Brightbean Pixonyx Ainyx Dablist	company_name ----- Du monde entier FISSA Fabrica Inter. Salchichas S.A. France restauration GROSELLA-Restaurante Laughing Bacchus Wine Cellars Lazy K Kountry Store North/South Paris spacialitus Pericles Comidas clusicas Princesa Isabel Vinhos Santu Gourmet Seven Seas Imports Spacialitus du monde The Big Cheese Vins et alcools Chevalier Roombo Fivebridge Fadeo Zooxo Thoughtbeat Fadeo Brightbean Pixonyx Ainyx Dablist	✓

Пройшов усі тести! ✓

Правильно

Бали за цю відповідь: 1,00/1,00.

Питання 6

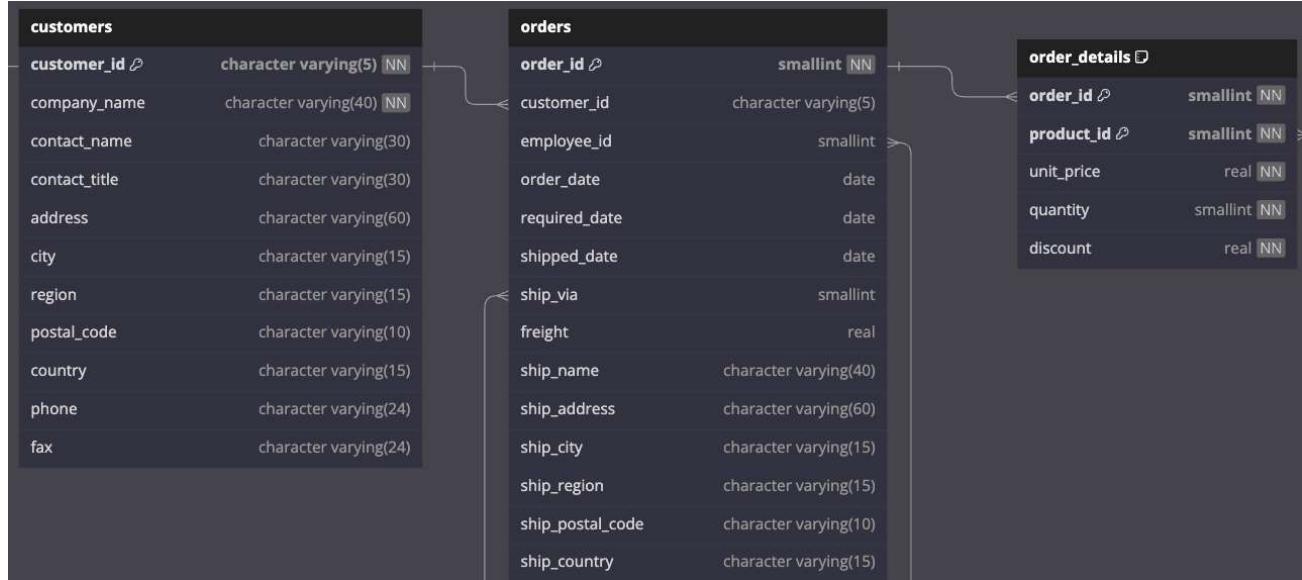
Правильно

Балів 1,00 з 1,00

We want to send all of our high-value customers a special VIP gift. We're defining high-value customers as those who've made at least 1 order with a total value (not including the discount) equal to \$10,000 or more. We only want to consider orders made in the year 1996 (for test use 2016, see note).

Use the alias **total_order_amount** for the calculated column. Order by the **total_order_amount**, in descending order.

Use the model:



Note. The test is running using SQLite Database that has the same structure, but slightly different information! You need to change the year from 1996 to 2016 when you copy your code from supabase. Also, a reminder that you need to use function **strftime('%Y', order_date)** to extract year from date instead of PostgreSQL **EXTRACT** when you copy your code from supabase.

For example:

Test	Result			
-- Testing with original db	customer_id	company_name	order_id	total_order_amount
	QUICK	QUICK-Stop	10865	17250.0
	SAVEA	Save-a-lot Markets	11030	16321.9
	HANAR	Hanari Carnes	10981	15810.0
	RATTC	Rattlesnake Canyon Grocery	10889	11380.0
	HUNGO	Hungry Owl All-Night Grocers	10897	10835.24

Відповідь: (penalty regime: 0 %)

```

1 | SELECT c.customer_id,
2 |     c.company_name,
3 |     o.order_id,
4 |     SUM(od.unit_price * od.quantity) AS total_order_amount
5 | FROM customers AS c
6 | JOIN orders AS o ON c.customer_id = o.customer_id
7 | JOIN order_details AS od ON o.order_id = od.order_id
8 | WHERE strftime('%Y', o.order_date) = '2016'
9 | GROUP BY c.customer_id, c.company_name, o.order_id
10 | HAVING total_order_amount >= 10000
11 | ORDER BY total_order_amount DESC;
12 |

```

	Test	Expected	Got	
✓	-- Testing with original db	<pre>customer_id company_name order_id total_order_amount ----- ----- QUICK QUICK-Stop 10865 17250.0 SAVEA Save-a-lot Markets 11030 16321.9 HANAR Hanari Carnes 10981 15810.0 RATTC Rattlesnake Canyon Grocery 10889 11380.0 HUNGO Hungry Owl All-Night Grocers 10897 10835.24</pre>	<pre>customer_id company_name order_id total_order_amount ----- ----- QUICK QUICK-Stop 10865 17250.0 SAVEA Save-a-lot Markets 11030 16321.9 HANAR Hanari Carnes 10981 15810.0 RATTC Rattlesnake Canyon Grocery 10889 11380.0 HUNGO Hungry Owl All-Night Grocers 10897 10835.24</pre>	✓
✓	-- Testing with extra row	<pre>customer_id company_name order_id total_order_amount ----- ----- SAVEA Save-a-lot Markets 11030 16321.9 HANAR Hanari Carnes 10981 15810.0 RATTC Rattlesnake Canyon Grocery 10889 11380.0 HUNGO Hungry Owl All-Night Grocers 10897 10835.24</pre>	<pre>customer_id company_name order_id total_order_amount ----- ----- SAVEA Save-a-lot Markets 11030 16321.9 HANAR Hanari Carnes 10981 15810.0 RATTC Rattlesnake Canyon Grocery 10889 11380.0 HUNGO Hungry Owl All-Night Grocers 10897 10835.24</pre>	✓ //

Пройшов усі тести! ✓

Правильно

Бали за цю відповідь: 1,00/1,00.

Питання 7

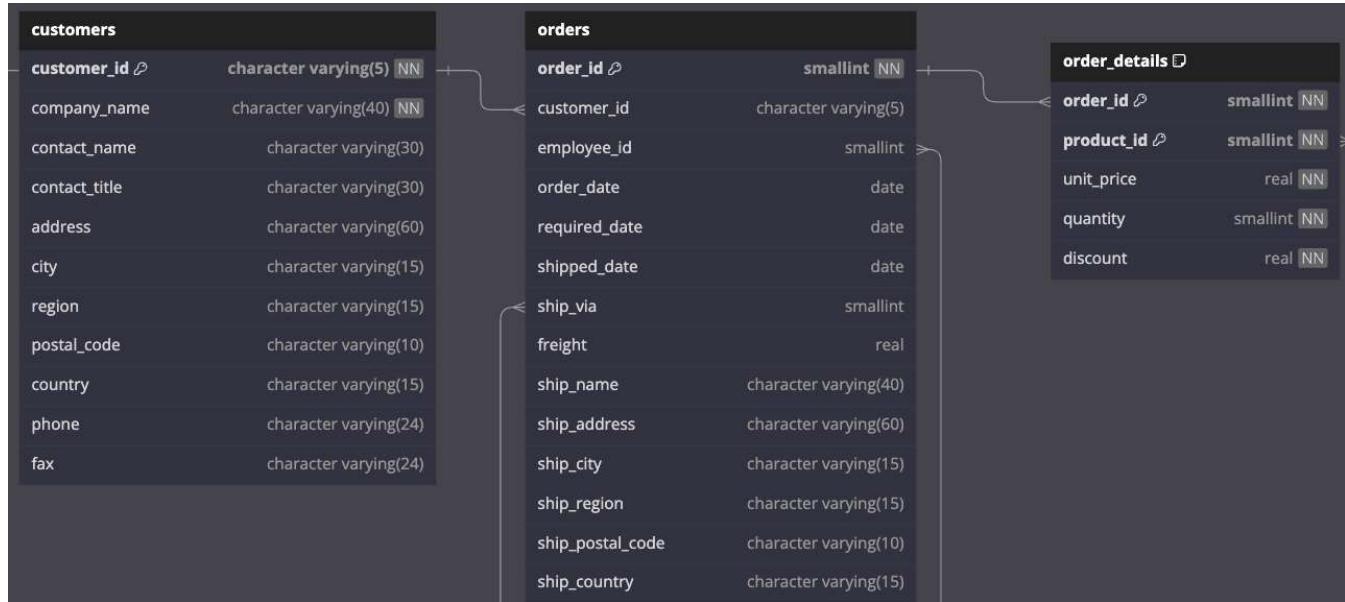
Правильно

Балів 1,00 з 1,00

We want to send all of our high-value customers a special VIP gift. We're defining high-value customers as those who have orders totaling \$15,000 or more (not including the discount) in 1996 (for test use 2016, see note).

Use the alias **total_order_amount** for the calculated column. Order by the **total_order_amount**, in descending order.

Use the model:



Note. The test is running using SQLite Database that has the same structure, but slightly different information! You need to change the year from **1996** to **2016** when you copy your code from **supabase**. Also, a reminder that you need to use function **strftime('%Y', order_date)** to extract year from date instead of PostgreSQL **EXTRACT** when you copy your code from **supabase**.

For example:

Test	Result		
-- Testing with original db	customer_id	company_name	total_order_amount
	-----	-----	-----
	SAVEA	Save-a-lot Markets	42806.25
	ERNSH	Ernst Handel	42598.9
	QUICK	QUICK-Stop	40526.99
	HANAR	Hanari Carnes	24238.05
	HUNGO	Hungry Owl All-Night Grocers	22796.34
	RATTC	Rattlesnake Canyon Grocery	21725.6
	FOLKO	Folk och fu HB	15973.85
	WHITC	White Clover Markets	15278.9

Відповідь: (penalty regime: 0 %)

```

1  SELECT c.customer_id,
2      c.company_name,
3      SUM(od.unit_price * od.quantity) AS total_order_amount
4  FROM customers AS c
5  JOIN orders AS o ON c.customer_id = o.customer_id
6  JOIN order_details AS od ON o.order_id = od.order_id
7  WHERE strftime('%Y', o.order_date) = '2016'
8  GROUP BY c.customer_id, c.company_name
9  HAVING total_order_amount >= 15000
10 ORDER BY total_order_amount DESC;
11

```

	Test	Expected	Got	
✓	-- Testing with original db	<pre>customer_id company_name total_order_amount ----- ----- SAVEA Save-a-lot Markets 42806.25 ERNSH Ernst Handel 42598.9 QUICK QUICK-Stop 40526.99 HANAR Hanari Carnes 24238.05 HUNGO Hungry Owl All-Night Grocers 22796.34 RATTC Rattlesnake Canyon Grocery 21725.6 FOLKO Folk och fu HB 15973.85 WHITC White Clover Markets 15278.9</pre>	<pre>customer_id company_name total_order_amount ----- ----- SAVEA Save-a-lot Markets 42806.25 ERNSH Ernst Handel 42598.9 QUICK QUICK-Stop 40526.99 HANAR Hanari Carnes 24238.05 HUNGO Hungry Owl All-Night Grocers 22796.34 RATTC Rattlesnake Canyon Grocery 21725.6 FOLKO Folk och fu HB 15973.85 WHITC White Clover Markets 15278.9</pre>	✓
✓	-- Testing with extra row	<pre>customer_id company_name total_order_amount ----- ----- SAVEA Save-a-lot Markets 42806.25 ERNSH Ernst Handel 42598.9 HANAR Hanari Carnes 24238.05 HUNGO Hungry Owl All-Night Grocers 22796.34 RATTC Rattlesnake Canyon Grocery 21725.6 FOLKO Folk och fu HB 15973.85 WHITC White Clover Markets 15278.9</pre>	<pre>customer_id company_name total_order_amount ----- ----- SAVEA Save-a-lot Markets 42806.25 ERNSH Ernst Handel 42598.9 HANAR Hanari Carnes 24238.05 HUNGO Hungry Owl All-Night Grocers 22796.34 RATTC Rattlesnake Canyon Grocery 21725.6 FOLKO Folk och fu HB 15973.85 WHITC White Clover Markets 15278.9</pre>	✓

Пройшов усі тести! ✓

Правильно

Бали за цю відповідь: 1,00/1,00.

Питання 8

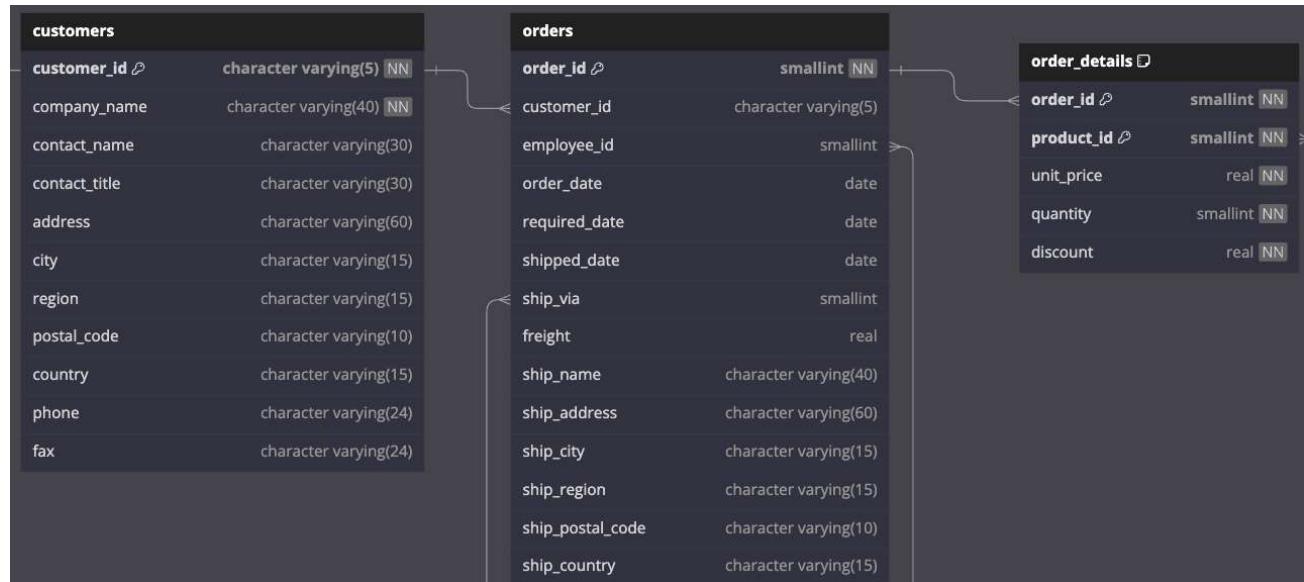
Правильно

Балів 1,00 з 1,00

We want to send all of our high-value customers a special VIP gift. We're defining high-value customers as orders totaling \$15,000 or more in 1996 (for test use 2016, see note). The result set should include the column **total_order_amount** with the total sum not including the discount, and the column **total_with_discount** with the total sum including the discount.

Order by **total_order_amount** in descending order.

Use the model:



Use `ROUND(<value>, 2)` function to round up **total_with_discount** values to the 2nd digit after the decimal point. See note.

Note. The test is running using SQLite Database that has the same structure, but slightly different information! You need to change the year from **1996** to **2016** when you copy your code from **supabase**.

Also, a reminder that you need to use function `strftime('%Y', order_date)` to extract year from date instead of PostgreSQL **EXTRACT** when you copy your code from **supabase**.

In **supabase** you might need to cast the **<value>** of the `ROUND` function to **numeric** type which is not needed in the test.

For example:

Test	Result			
-- Testing with original db	customer_id	company_name	total_order_amount	total_with_discount
	SAVEA	Save-a-lot Markets	42806.25	36310.11
	ERNSH	Ernst Handel	42598.9	41210.65
	QUICK	QUICK-Stop	40526.99	37217.32
	HANAR	Hanari Carnes	24238.05	23821.2
	HUNGO	Hungry Owl All-Night Grocers	22796.34	20402.12
	RATTC	Rattlesnake Canyon Grocery	21725.6	21238.27
	WHITC	White Clover Markets	15278.9	15278.9

Відповідь: (penalty regime: 0 %)

```

1  SELECT c.customer_id,
2      c.company_name,
3      SUM(od.unit_price * od.quantity) AS total_order_amount,
4      ROUND(SUM(od.unit_price * od.quantity * (1 - od.discount)), 2) AS total_with_discount
5  FROM customers AS c
6  JOIN orders AS o ON c.customer_id = o.customer_id
7  JOIN order_details AS od ON o.order_id = od.order_id
8  WHERE strftime('%Y', o.order_date) = '2016'
9  GROUP BY c.customer_id, c.company_name
10 HAVING total_with_discount >= 15000
11 ORDER BY total_order_amount DESC;
12

```

	Test	Expected	Got	
<input checked="" type="checkbox"/>	-- Testing with original db	<pre>customer_id company_name total_order_amount total_with_discount -----</pre> <pre>SAVEA Save-a-lot Markets 42806.25 36310.11 ERNSH Ernst Handel 42598.9 41210.65 QUICK QUICK-Stop 40526.99 37217.32 HANAR Hanari Carnes 24238.05 23821.2 HUNGO Hungry Owl All-Night Grocers 22796.34 20402.12 RATTC Rattlesnake Canyon Grocery 21725.6 21238.27 WHITC White Clover Markets 15278.9 15278.9</pre>	<pre>customer_id company_name total_order_amount total_with_discount -----</pre> <pre>SAVEA Save-a-lot Markets 42806.25 36310.11 ERNSH Ernst Handel 42598.9 41210.65 QUICK QUICK-Stop 40526.99 37217.32 HANAR Hanari Carnes 24238.05 23821.2 HUNGO Hungry Owl All-Night Grocers 22796.34 20402.12 RATTC Rattlesnake Canyon Grocery 21725.6 21238.27 WHITC White Clover Markets 15278.9 15278.9</pre>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/>	-- Testing with extra row	<pre>customer_id company_name total_order_amount total_with_discount -----</pre> <pre>SAVEA Save-a-lot Markets 42806.25 36310.11 ERNSH Ernst Handel 42598.9 41210.65 HANAR Hanari Carnes 24238.05 23821.2 HUNGO Hungry Owl All-Night Grocers 22796.34 20402.12 RATTC Rattlesnake Canyon Grocery 21725.6 21238.27 WHITC White Clover Markets 15278.9 15278.9</pre>	<pre>customer_id company_name total_order_amount total_with_discount -----</pre> <pre>SAVEA Save-a-lot Markets 42806.25 36310.11 ERNSH Ernst Handel 42598.9 41210.65 HANAR Hanari Carnes 24238.05 23821.2 HUNGO Hungry Owl All-Night Grocers 22796.34 20402.12 RATTC Rattlesnake Canyon Grocery 21725.6 21238.27 WHITC White Clover Markets 15278.9 15278.9</pre>	<input checked="" type="checkbox"/>

Пройшов усі тести! Правильно

Бали за цю відповідь: 1,00/1,00.

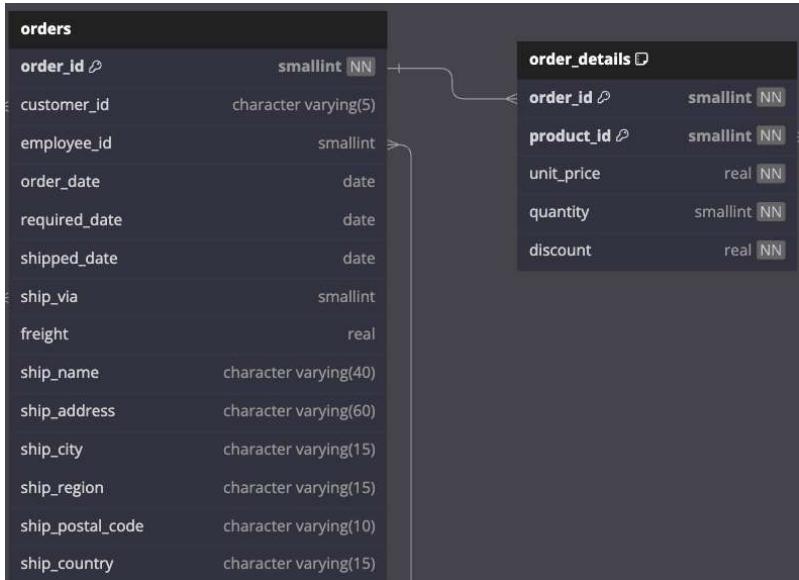
Питання 9

Правильно

Балів 1,00 з 1,00

The Northwind mobile app developers are testing an app that customers will use to show orders. In order to make sure that even the largest orders will show up correctly on the app, they'd like some samples of orders that have lots of individual line items (entries in **order_details**). Show the 10 orders with the most line items, in order of total line items.

Use the model:



Note. Using **orders** and **order_details**, you'll use *Group by* and *count()* functionality.

Use the alias **total_order_lines** for the calculated column.

Use

```
select
<body of the select statement>
```

```
limit 10;
```

to limit the number of rows in the result set.

For example:

Test	Result
-- Testing with original db	order_id total_order_lines
	----- -----
	11077 25
	10657 6
	10847 6
	10979 6
	10273 5
	10294 5
	10309 5
	10324 5
	10325 5
	10337 5

Відповідь: (penalty regime: 0 %)

```
1 | SELECT o.order_id,
2 |     COUNT(od.order_id) AS total_order_lines
3 | FROM orders AS o
4 | JOIN order_details AS od ON o.order_id = od.order_id
5 | GROUP BY o.order_id
6 | ORDER BY total_order_lines DESC
7 | LIMIT 10;
8 |
```

Test	Expected	Got	
<input checked="" type="checkbox"/> -- Testing with original db	<pre>order_id total_order_lines ----- ----- 11077 25 10657 6 10847 6 10979 6 10273 5 10294 5 10309 5 10324 5 10325 5 10337 5</pre>	<pre>order_id total_order_lines ----- ----- 11077 25 10657 6 10847 6 10979 6 10273 5 10294 5 10309 5 10324 5 10325 5 10337 5</pre>	<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> -- Testing with extra row	<pre>order_id total_order_lines ----- ----- 11077 25 10337 11 10847 6 10979 6 10273 5 10294 5 10309 5 10324 5 10325 5 10360 5</pre>	<pre>order_id total_order_lines ----- ----- 11077 25 10337 11 10847 6 10979 6 10273 5 10294 5 10309 5 10324 5 10325 5 10360 5</pre>	<input checked="" type="checkbox"/>

Пройшов усі тести!

Правильно

Бали за цю відповідь: 1,00/1,00.

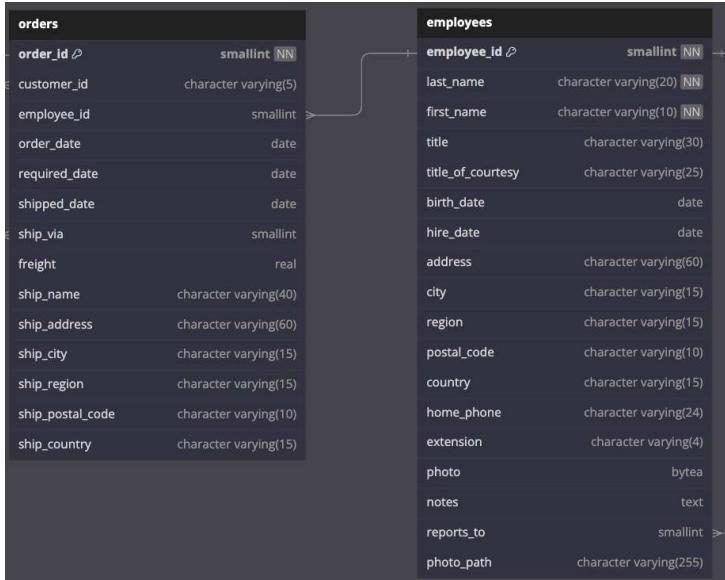
Питання 10

Правильно

Балів 1,00 з 1,00

Some sales people have more orders arriving late than others. Maybe they're not following up on the order process, and need more training. Which sales people have the most orders arriving late?

Use the model:



Note. To determine which orders are late, you can use a combination of the **required_date** and **shipped_date**. It's not exact, but if **shipped_date** is actually after **required_date**, you can be sure it's late.

Use the alias **total_late_orders** for the calculated column.

You'll need to join to the **employees** table to get the last name, and also add COUNT to show the **total_late_orders**.

For example:

Test	Result
-- Testing with original db	employee_id last_name total_late_orders
	----- ----- -----
	4 Peacock 10
	3 Leverling 5
	2 Fuller 4
	7 King 4
	8 Callahan 4
	9 Dodsworth 4
	6 Suyama 3
	1 Davolio 2
	5 Buchanan 1

Відповідь: (penalty regime: 0 %)

```

1 | SELECT e.employee_id, e.last_name,
2 |     COUNT(o.order_id) AS total_late_orders
3 | FROM orders AS o
4 | JOIN employees AS e ON o.employee_id = e.employee_id
5 | WHERE o.shipped_date > o.required_date
6 | GROUP BY e.employee_id, e.last_name
7 | ORDER BY total_late_orders DESC;
8 |

```

	Test	Expected	Got	
✓	-- Testing with original db	<pre>employee_id last_name total_late_orders ----- ----- 4 Peacock 10 3 Leverling 5 2 Fuller 4 7 King 4 8 Callahan 4 9 Dodsworth 4 6 Suyama 3 1 Davolio 2 5 Buchanan 1</pre>	<pre>employee_id last_name total_late_orders ----- ----- 4 Peacock 10 3 Leverling 5 2 Fuller 4 7 King 4 8 Callahan 4 9 Dodsworth 4 6 Suyama 3 1 Davolio 2 5 Buchanan 1</pre>	✓
✓	-- Testing with extra row	<pre>employee_id last_name total_late_orders ----- ----- 3 Leverling 5 2 Fuller 4 7 King 4 8 Callahan 4 9 Dodsworth 4 6 Suyama 3 1 Davolio 2 5 Buchanan 1</pre>	<pre>employee_id last_name total_late_orders ----- ----- 3 Leverling 5 2 Fuller 4 7 King 4 8 Callahan 4 9 Dodsworth 4 6 Suyama 3 1 Davolio 2 5 Buchanan 1</pre>	✓ //

Пройшов усі тести! ✓

Правильно

Бали за цю відповідь: 1,00/1,00.